Eyal Orbach – ID 015369317
Daniel Juravski – ID 206082323

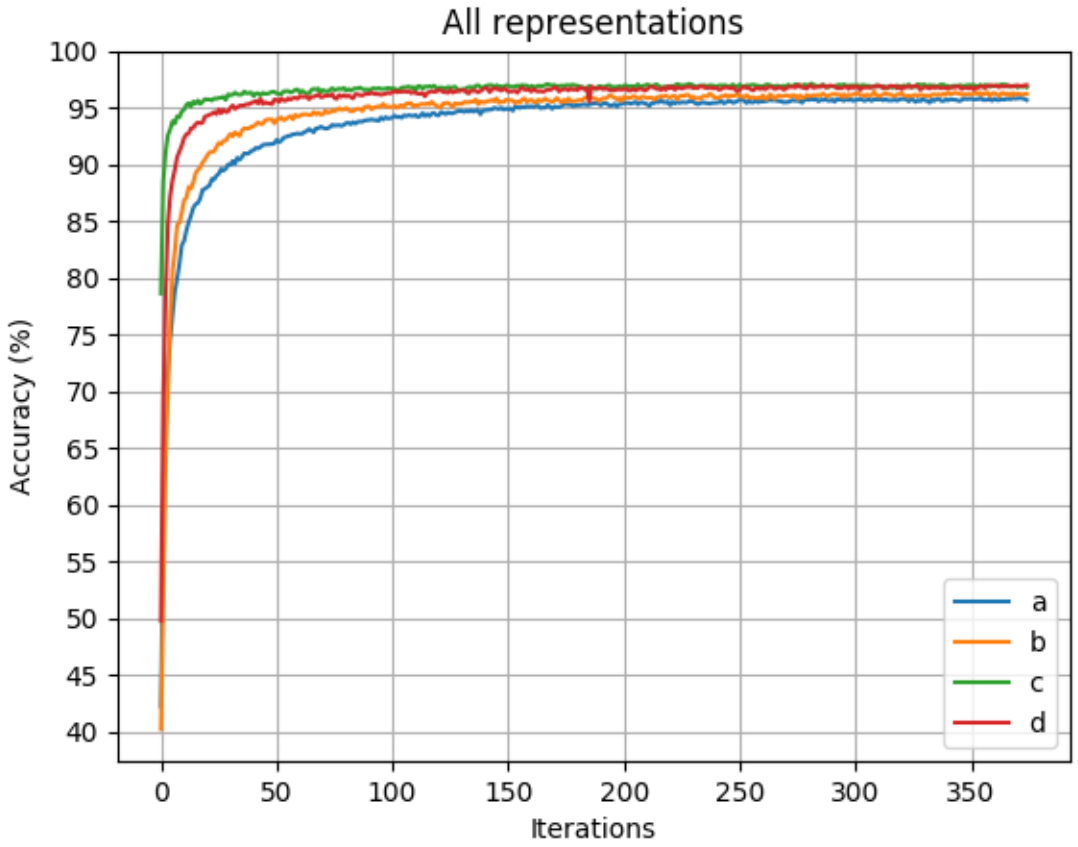
Our model:

- BiLSTM:
  - First layer:
    - INPUT_DIM = 50
    - STATE_DIM = 50
  - Second layer:
    - INPUT_DIM = 100
    - STATE_DIM = 50
- Embedding layer:
  - ROW_SIZE = VOCABLURY_SIZE
  - COL_SIZE = 50
- Linear layer:
  - INPUT_DIM = 100
  - OUTPUT_DIM = TAG_SET_SIZE (=45)
- Optimizer: AdamTrainer
- Learning Rate: default of 0.001

We evaluated our model over 2 data sets: 'POS' and 'NER.

The results that we get were as we expected, the (c) representation achieved the best accuracy mainly because of the sub-words representation (the pre-trained embeddings just speed up the learning). Those sub-words were the perfect features for representing the words (even words that never were seen before) and 'understanding' the word behavior.
Right after that representation, we got the (d) representation that is a concatenation of (a) and (b) followed by a linear layer. That method is quite similar to the sub-words method, but on this it is about single chars and whole words. On the one hand, the chars do not always can get the word's behavior, on the other hands, neither the words' – because the test data includes words that we might not seen before.
Due to those reasons, we got the (b) representation right after, because we can represent every word with characters embedding, and the last is (d), once again – there can be many words that we didn't learned in the train, and now just replaced by the 'unk' embedding.

POS Data:



NER Data: